

Ôn tập MIPS

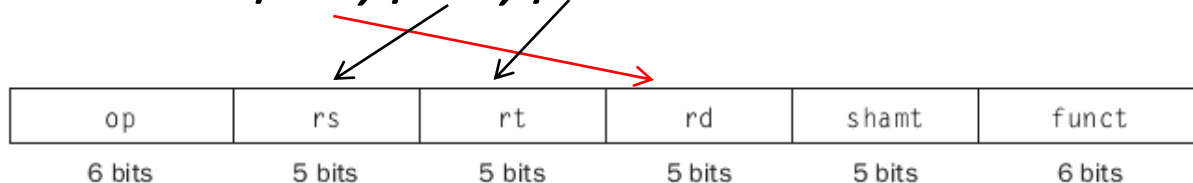
Th.Si: Ha Le Hoai Trung

Nội dung

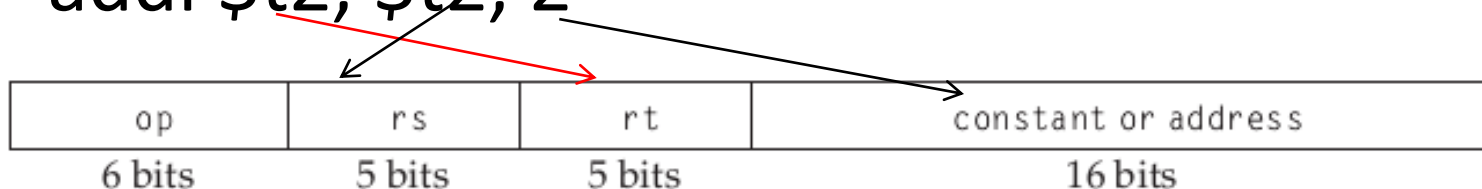
- Chuyển mã assembly MIPS sang dạng mã máy MIPS
- Chuyển dạng mã máy MIPS sang dạng mã assembly MIPS
- Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS
 - Lệnh cộng, trừ, nhân, chia
 - Mảng
 - Control: if, for, while, ...
 - Function: parameter (register: save, not save), return, stack pointer, return address.
- Cho 1 đoạn lệnh assembly MIPS, đọc kết quả của 1 thanh ghi.

Chuyển mã assembly MIPS sang dạng mã máy MIPS

- ***add \$t0,\$s1,\$s2***



- Xác định loại lệnh
 - Thanh ghi: `add`, `sub`, `or`, ...
 - Giá trị tức thời: `lw`, `sw`, `addi`, `subi`, ...
- ***addi \$t2, \$t2, 2***



Chuyển mã assembly MIPS sang dạng mã máy MIPS

- **add \$t0, \$s1, \$s2**
- **sub \$s0, \$s2, \$s3**
- **sll \$s2, \$s4, 12**
- **addi \$t0, \$t0, -1**
- **lw \$s1, 0(\$s0)**
- **lw \$s1, 32(\$s0)**
- **sw \$a0, 16(\$t0)**
- **beq \$t0, \$0, 50**

Chuyển dạng mã máy MIPS sang dạng mã assembly MIPS

- Binary
- 6 bit trọng số cao – opcode → tra bảng trong sách
- R (thanh ghi), I (tức thời), J (nhảy)
- Bao nhiêu thanh ghi ??
- Giá trị tức thời lưu trữ theo dạng bù 2. Trọng số cao nhất trong dạng bù 2
 - 0: số dương
 - 1: số âm

Chuyển dạng mã máy MIPS sang dạng mã assembly MIPS

- 00001025_{hex}
- 0005402A_{hex}
- 11000003_{hex}
- 00441020_{hex}
- 20A5FFFF_{hex}
- 08100001_{hex}

Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

- Biểu thức.
 - Chỉ số thanh ghi
 - Loại lệnh R, I, J

- Ví dụ

$$f = g - h;$$

$$f = g + (h - 5);$$

$$f = g * 2;$$

Biết f , g , h nằm trong các thanh ghi $t1$, $t2$, $t3$.

Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

- Mảng
 - Chỉ số mảng * 4 $\rightarrow$ xác định khoảng cách từ địa chỉ base đến địa chỉ ô nhớ cần xác định.
 - Địa chỉ base + khoảng cách.

- Ví dụ

$f = g + h + B[4];$

$B[8] = A[i-j];$

$f = A[B[g]*4];$

Assume that the variables $f, g, h, i,$ and j are assigned to registers $\$s0, \$s1, \$s2, \$s3,$ and $\$s4$, respectively.

Assume that the base address of the arrays A and B are in registers $\$s6$ and $\$s7$, respectively.

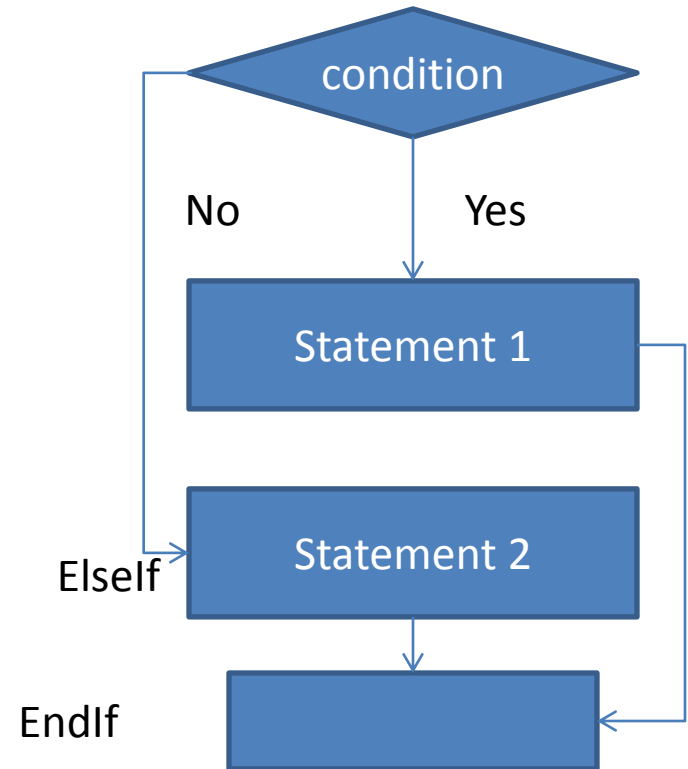
Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

- Control: if, while, for, ...

```
if( condition){  
    Statement 1;  
} else {  
    Statement 2;  
}
```

- Ví dụ

```
if ((x + y) < 3)  
    x = x + 5;  
else  
    y = y + 4;
```



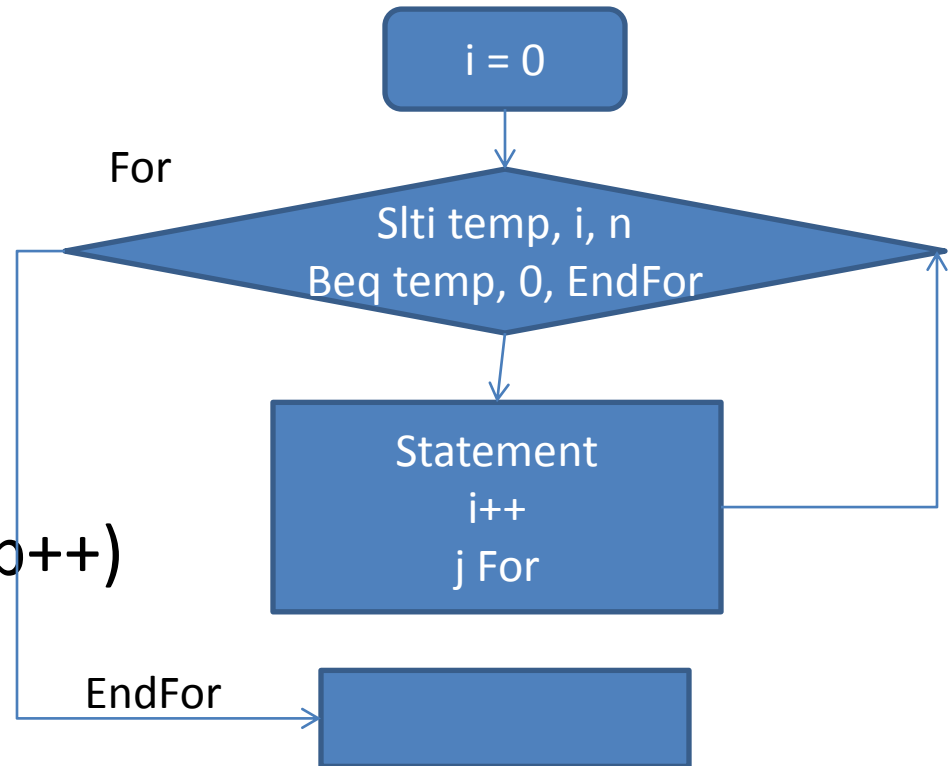
Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

- Control: if, while, for, ...

```
for (int i = 0; i < n; i++)  
{  
    Statement;  
}
```

- Ví dụ:

```
for (a = b = 0 ; b != 10 ; b++)  
    a += b;
```



A, b sử dụng thanh ghi t1, t2.

Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

```
int i, s, A[10];  
for (s = i = 0 ; i < 10 ; i++)  
    s = s + A[i];
```

i,s sử dụng thanh ghi a1, a0.

Mảng A nằm trong thanh ghi t3.

```
int s, i, a[10];  
for (s=0, i = a+9 ; i >= a ; i--)  
    s += i;
```

Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

```
for (i = 2; i<10; i++)  
    A[i]= B[A[i-2]];
```

Giả sử biến i được gán vào thanh ghi \$s0, địa chỉ nền của mảng A và B lần lượt được lưu trong thanh ghi \$s1 và \$s2.

Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

2 vòng loop:

i = 10;

do {

temp = 10;

do {

B += 2;

temp = temp - 1;

} while (temp > 0)

i = i - 1;

} while (i > 0);

Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

- **Function:**
 - Put **parameters** in a place where the procedure can access them. $\$a0 - \$a3$ (stack pointer)
 - Transfer **control** to the procedure. *jal name_procedure*
 - Acquire the **storage resources** needed for the procedure. Lưu các tham số cần thiết và địa chỉ trả về cho hàm vào trong stack, stack pointer sẽ giảm.
 - Put the **result** value in a place where the **calling program** can access it. $\$v0-\$v1$: **two value registers** in which to return values
 - **Return** control to the point of origin. *jr \$ra*: **one return address register** to return to the point of origin – địa chỉ này lấy từ stack pointer

Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

- Ví dụ:

```
int positive(int a, int b) {  
    if (addit(a, b) > 0)  
        return 1;  
    else  
        return 0;  
}  
int addit(int a, int b) {  
    return a+b;  
}
```

Bài tập

- Chuyển mã assembly MIPS sang dạng mã máy MIPS

and \$s1,\$t2,\$s3

or \$t1,\$s2,\$t3

andi \$a1,\$a2,10

ori \$a1,\$t2,10

sll \$s6,\$t5,10

srl \$s5,\$t6,10

lw \$s0,10(\$s2)

sw \$s1,10(\$s2)

beq \$s1,\$s2,10

add \$t0, \$t2, \$t1

addi \$t0, \$t2, -5

sub \$t5, \$s3, \$t3

bne \$s1,\$s2,10

slt \$t1,\$t2,\$3

Bài tập

- Chuyển dạng mã máy MIPS sang dạng mã assembly MIPS:

01494020

0171502A

2149FFFF

024B4825

1632000A

Bài tập

- Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

$$a = b * c + 3$$

$$a = b * 8 - a$$

$$a = -b - c + 120.$$

Bài tập

- Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

```
if( (a - b) > 0){  
    if (a > 5) {  
        a = a + 4;  
    }  
else {  
    b = b + 10;  
}
```

Bài tập

- Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

```
for (int i = 0; i < n; i++){  
    for (int j = n-1; j >= 0; j--) {  
        A[i] = B[j];  
    }  
}
```

Biết các biến i, j, n nằm trong các thanh ghi $\$t1, \$t2, \$t3$. Các mảng A, B nằm trong thanh ghi $\$s1, \$s2$.

Bài tập

- Chuyển dạng mã ngôn ngữ cấp cao sang dạng mã assembly MIPS

```
int countSoNguyenTo(int a) {  
    int count = 0;  
    for (int i = 0; i < a; i++){  
        if (songuyento(i) == 1)  
            count ++;  
    }  
    return count;  
}  
  
int songuyento(int a) {  
    for (int i = 0; i < a/2; i++){  
        if (a % i == 0){  
            return 1;  
        }  
    }  
    return 0;  
}
```